

Legalábbis amikor pár hónappal ezelőtt a BIT-LET szerkesztője felkereste elképzelésével a KISZ Központi Bizottság illetékes osztályvezetőjét, maga sem igazán hitt a dologban. Olyan technicizált újság elkészítésére, kipróbálására akarta ugyanis rábeszélni a középiskolásokkal foglalkozó vezetőket, amelyről ma Magyarországon még csak álmodnak néhányan, de amelynek megvalósításáról hallani sem akarnak az illetékesek, mert ha a technika meg is lenne hozzá, teljesen új szemléletet, munkastílust követelne a munkatársaktól. Az ötlet tetszett, az anyagi feltételek megteremthetők voltak, így hát beindult a munka. Nos a csoda megtörtént, április 30. és május 3. között minden tökéletesen működött. Hogy pontosan mi történt és hogyan, ezt próbáljuk meg összefoglalni az alábbi kis naplóval.

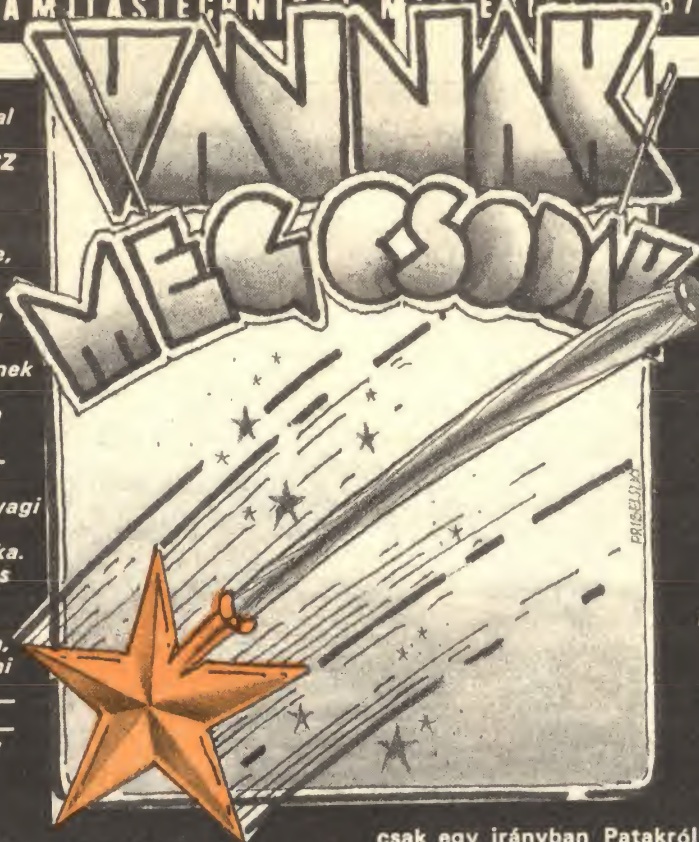
1986. OKTÓBER

E sorok szerzője kitalálja, hogy az Országos Diáknapok egyidőben hét városban zajló eseményeiről érdemes lenne megjelentetni egy olyan lapot, amelybe minden helyszínről érkeznek tudósítások, s amely másnap reggelre mindenhova odaér. Az illetékeseknek elmondja az alapötletet, akik megszeretik az elképzelést, s megbízzák, hogy dolgozza ki a megvalósítás módját.

1986. NOVEMBER

Egy régi ismerősömtől véletlenül megtudom, hogy rövidesen megindul a gyártása egy hazai fejlesztésű akusztikus modemnek, amely C64-eshez is kapcsolható. A cég – a neve sem titok – a COOPINFORM nem zárkózik el attól, hogy segítségül legyen, s mintegy tíz db modemet kölcsönadjon a néhány napos eseményre.

(Megjegyzés: akusztikus modemnek hívják azokat a berendezéseket, amelyekre egyszerűen ráhelyezhetjük a telefonkagylót, s ha előzőleg a berendezés kimenetét rácsatlakoztattuk mondjuk egy Commodore 64-esre, s a megfelelő programot is lefuttattuk rajta, akkor bármilyen programot, vagy szövegfile-t elküldhetünk rajta a telefonvonal túlsó végén lévő hasonló berendezéssel rendelkező partnernek.)



1986. DECEMBER

A modemek jól vizsgáznak a BIT-LET Karácsony bemutatóján.

1987. JANUÁR

A dolgok nem haladnak. A modemekből az első két prototípuson kívül nincs több. Az ígért próbadarabokat az istennek sem kapja meg a most már háromfősre bővült stáb. Közben azért megindul a szervezés, ha lassan is. Ígérek, hogy lesz a dologhoz olcsó nyomda. A diáknapokat rendező városokban várakozó állásponton vannak – szervezzenek saját újságot vagy sem...

Innentől kezdve sokáig nem történik semmi, illetve semmi olyan, ami a BIT-LET-híveket izgathatná. Ezért hát ugorjunk...

1987. MÁRCIUS

Megérkezik az első nullszériás modemek. Az egyik munkatárs leutazik Sárospatakra, ahonnan sikerrel bejelentkezik, de a posta pillanatok alatt szétbontja a vonalat. Az újabb hívásnál

csak egy irányban Patakról Budapestre működik az átvitel, odalent semmi sem vehető. Két nappal később a modem fejlesztője megvizsgálja a berendezést, s megállapítja, hogy apró konstrukciós hibát követett el, amely lehetővé tette, hogy az erőteljesen behelyezett telefonkagyló szétnyomja a modem mikrofonját. A konstrukción apró változtatást eszközöl.

1987. ÁPRILIS 1.

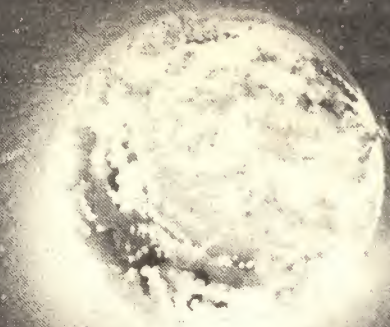
Utolsó, már így is elkésett pillanat, amikor dönteni kell. Belevágunk-e a dologba vagy sem? Működnek majd a modemek? A javított változat bírja majd a háromnapos strapát? Közben mikor kiderül, hogy az egész rendezvény a május elejei ünnepnapokra esik, minden nyomda visszalép a munkától. Szaladgálás, telefonálás, végül is döntés: csináljuk! Nyomda is akad, a modemek is szinte hibátlanul működnek az újabb próbák során.

FOLYTATÁS A 22. OLDALON

BELÜLRŐL

- 18 **Híroldal** – amelyben megismerkedhetnek a világ leggyorsabb hordozható személyi számítógépével a MultiSpeed-del.
- 20 **Programajánlat** – egy monitor a Tévé-computerre.
- 22 **Vannak még csodák** – amit már itt a címdoldalon elkezdtek olvasni, befejezhetik néhány oldallal arrébb...
- 24 **Atari kör** – a körben ezúttal egy komolyabb programot olvashatnak az Atari sprite-jairól, valamint néhány címet a gép belsejéből.
- 26 **Almavírágzás** – az alma azaz az Apple legújabb gépe méltán vívta ki a szakma érdeklődését. A Macintosh II. megjelenése alkalmából áttekintjük a cég múltját és jelenét.
- 28 **Programajánlat** – karakter editor a C 16-ra. Ennek a programnak a fő erénye hogy rendkívül rövid és mégis tudja, amit tudnia kell.
- 29 **Sorvezető** – ezúttal is oktatást segítő anyag a valószínűségi számításról.
- 30 **Könyvmoly** – új könyvekkel, s a C64/128 zenekönyvről szóló kritikával
- 31 **Posta és cserebere** – a kettő ezúttal kicsit keveredik, mert a cserebere olvasói is levelekben tesznek panaszt egymásra.
- 32 **Atari nyerő** – három hónapos pályázatunk utolsó feladata, s már az első megoldása is...

HÍROLDAL



ROBOTFORGALOM

Egyesült államokbeli szakemberek egy csoportja 2010-re prognosztizálja az automati-
kus autóforgalom első megoldásának létre-
hozását. Az elképzelések szerint speciálisan
megtervezett és kivitelezett autópályákon
mintegy kétszázharminc kilométeres sebes-
séggel, de nem utas nélküli gépkocsik. A köz-
lekedő járművek minden mozgulatát egy
központi óriás számítógép irányítaná, tökéle-
tesen figyelembe véve az utakat, azok le-
terhelését, az egyes járművek úticélját. A jós-
lat vitatói felhívják a figyelmet a rendszer
megvalósításához szükséges különleges út-
hálózat kiépítésének hatalmas költségeire, és
felteszik a kérdést: mi lesz azokkal az autó-
tulajdonosokkal, akik kifejezetten szeretnek
vezetni.

SZOBOL ÉRT

Újabb csúcst ért el az amerikai IBM cég.
Ezúttal a beszédfelismerő számítógépek ver-
senyében. Sikert ért el a hallás útján mintegy
számítógépet, amely hallás útján mintegy
húsz ezer szót képes felismerni és kiírni a kép-
ernyőre. Mielőtt azonban produkcióját meg-
kezdhetné, szükséges, hogy megismerkedjen
kezelőjének hangjával, kiejtésével, beszéd-
ritmusával. Ezért az illető személynek egy
húszperces, speciális szöveget kell beolvas-
nia a gép mikrofonjába. Az ismerkedést kö-
vetően a húsz ezer szó bármelyikét felismeri
a számítógép. Az IBM továbbra sem ül az
elért siker babérait: megkezdte egy olyan
számítógép kifejlesztését, amely nemcsak
egyes szavakat ért, hanem a folyamatos em-
beri beszédet is.

SZIMULTÁN A GÉPEKKEL

A közelmúltban Bilek István nemzetközi
sakk különféle szimultán játszott tizen-
kilenc szerepeltek a világ jelenlegi legjobbjai
között tizenegy partit nyert, négyet el-
vesztett, négy pedig döntetlen lett, ami igen
szisztematikus eredmény – a számítógépek
István – Kaszparov világ bajnok és két hol-
land nagymester után – negyediként vál-
lalkozott számítógépek elleni szimultán já-
téka. A nagymester nem is emlékszik, mikor
szenvetett négy vereséget egy szimultánon.
Bilek István közvetlenül a számítógépekkel
– tehát sakktáblák közbeiktatása nélkül –
játszott, ami hátrányt jelentett a számára.

SZUPERCHIP

Az amerikai TRW cég huszonnyolcmillió
kapcsolási elemről álló szuper integrált
áramkör elkészítésén dolgozik. Az áramkör-
ben nagyon sok redundáns cella lesz és így
már a gyártás során automatikusan ki lehet
kerülni a hibás cellákat. Ugyanakkor a szu-
perchip mindig maga választ pótállást a re-
dundáns tartalékból. Az úgynevezett recon-
figuráció az áramköri elem hardverében el-
helyezett program szerint megy majd végbe.

VEVŐ A TASZSZ IS

Új típusú mátrixnyomtató gyártását kezdték
meg a székesfehérvári Videoton Elektronikai
Vállalatnál. A számítástechnikai gyárban már
elkészült az első kétszázötven darab az új
típusú mátrixnyomtatóból, amely a korábbi-
aknál szebb betűtípusokat ír. Az idén ezret
készítenek belőle, főként a hazai felhasználó-
lók részére. Kétszázötvenet a Szovjetunióba
exportálnak, ahol a TASZSZ hírügynökség
tudósítói munkahelyeit szerelik fel velük.

LSI-CENTRUMOK

A japán Toshiba cég eddig az NSZK-
ban és Nagy-Britanniában hozott létre
LSI áramkör-tervező központot. Úgy ter-
vezik, hogy két éven belül további négy
ilyen centrumot telepítenek Európába.
A tervező központok feladata az lesz,
hogy berendezésorientált áramkörökkel
lássák el az európai, mikroelektronikai
készülégyártókat. Az európai LSI cent-
rumok műholdas összeköttetésben van-
nak a Tokió melletti főközponttal.

BALESETELEMZŐ

Az angliai Bowring cég balesetelemző,
számítógépes programrendszert rögzíti a
zott ki. A Microsate rendszert rögzíti a
baleseti sérüléseket, az eredményeket
szükséges elemzéseket, az eredményeket
táblázatosan vagy grafikonokon is ábrá-
zolja. Segítségével feltárhatók azok az
általános körülmények, jelenségek, okok,
amelyek súlyos balesetekhez vezetnek.
A Microsate program IBM kompatibilis
rendszeren futtatható.

TAJVANI EXPORT

Mintegy kétmilliárd dollár értékben szállított
Tajvan a múlt évben számítógépeket és szá-
mlástechnikai eszközöket a világpiacra. Ez
az érték hatvan százalékkal múlta felül a
tavalyelőtti exportot. A fejlődés üteme óriási.
Mindössze hat éve annak, hogy az ország-
ban megkezdtek a számítástechnikai ipar
kiépítését. 1982-ben kezdtek meg az export-
tot, ami ma már a teljes tajvani kivitelben
huszonöt százalékot képvisel. Fő vásárló-
Nyugat-Európa, melynek piacáról fokozato-
san kisorsítják az amerikai és japán cégeket.

CHIPSZALLÍTÁS

Az áramköri morzsák vagy chippek szállítás
közben kétféleképpen is megsérülhetnek. Az
egyikféle a mechanikai sérülés, a másik
pedig az esetleg bekövetkező elektrostatikus
kiszülés, amely az áramkör belső felépítését
teszi tönkre. Most egy NSZK-beli cég Ultra-
mid néven olyan poliamid műanyagfóliát
fejlesztett ki, amely szilárdsága, mérettar-
sága és villamos szigetelési jellemzői alapján
mindkétféle veszélytől szemben jól védi a
chipeket, míg eljutnak előállítási helyüktől a
berendezésgyártó megrendelő üzemébe.

PROGRAMMA. IÁNLAT:

TV COMPUTER -
MONITOR

A program rendeltetése

A program egy gépi kód beírását megkönnyítő program, mely editor funkciókkal van ellátva. Megjegyzem, hogy ez csak egy monitor, tehát assemblerből nem fordít gépi kódra.

A program beírása

Először is írjuk be az 1. listát és indítsuk el. (Nagyon figyeljünk a csillagok számára!)

Ezután írjuk be a 2. listát és mielőtt elindítanánk, a programot másoljuk ki a:

POKE5895,255:SAVE"EDIMONI":
VERIFY parancssal.

Mint látjuk a programban nagyon kevés a szóköz (de, ahol szóköz van, annak ott is kell lenni, egyébként az interpreter nem érti meg), erre csak a kisebb helyfoglalás miatt van szükség.

Megjegyzés: Az 525-ös sorban a macskakörök közé ALT+D-t kell tenni

526-osban ALT+H-t

527-esben ALT+B-t

Felhasználás

Indítsuk el a programot. Megjelenik a fejléc és a program kérdezi a kezdőcímet. Ekkor megjelenik egy lista. Elöl mindig a byte sorszáma, mögötte pedig a byte tartalma látható. A listázást ezután a joystick-vel véggezhetjük fel-le, vagy új kezdő címet írhatunk be a hatványozás jel megnyomására. Középen egy kérdőjel jelenik meg, mely mindig az aktuális byte-ot mutatja. A mögötte lévő villogó "D", "H", vagy "B" azt mutatja, hogy éppen decimális, hexadecimális vagy bináris üzemmódban vagyunk. Ez azt jelenti, hogy decimálisan, hexadecimálisan vagy binárisan akarjuk az adatokat bevinni. Az üzemmódok az ALT és a D, H, B-vel válthatók. Lehet 255-nél nagyobb és lehet negatív számokat is bevinni, a program ezeket két byte-ra bontja, illetve kettes komplementbe váltja.

A parancsszavakat és hexadecimális számokat elég a kezdőbetűkkel begépelni (lehet kicsi, nagy is).

Az editor gombok közül egyedül a DEL működik. A számok vagy parancsok után értelemszerűen RETURN kell. A RETURN üres megnyomása a byte törlését jelenti.

A kezdőcímet nem lehet kisebb mint 10000 és nem lehet nagyobb mint 32767. A hibát mindig hangjelzés kíséri.

Fontos: A gépi kódú program végéről mindig tudnia kell a programnak, egyébként nem működnek az editor parancsok. Ezzel olyankor nincs gond, amikor a programot beírjuk vagy be-

töltjük. Ha viszont egy, már a tárban lévő programot akarunk editálni, akkor a monitor betöltése után a program utolsó byte-ját újból beírjuk. Ugyanis a programban van egy változó, amely mindig a legmagasabb címet jegyzi meg, és ilyen újból való beírásakor automatikusan tárolódik. Az editor parancsokhoz ez a legszükségesebb változó.

Ezt a kényelmetlenséget ellensúlyozza az, hogy a program vége akkor is megőrződik, ha meleg RESET-elünk és újra indítjuk a monitort.

```

1 ! Cím: *****HOSSZ: **KÍM: **
2 ! SAVE *****LOAD *****INS *****DEL ****
3 ! *****
4 ! AAA
10 FOR I=6685 TO 6702: READ A: POKE I,A: NEXT: ! SAVE
20 FOR I=6708 TO 6725: READ A: POKE I,A: NEXT: ! LOAD
30 FOR I=6730 TO 6732: READ A: POKE I,A: NEXT: ! INS
40 FOR I=6740 TO 6742: READ A: POKE I,A: NEXT: ! DEL
50 FOR I=6750 TO 6755: READ A: POKE I,A: NEXT: ! ATV.
60 !
100 DATA 17,248,25 : ! LD DE,6648
110 DATA 247,83 : ! RST 48,83
120 DATA 237,91,16,26: ! LD DE,(6672)
130 DATA 237,75,9,26 : ! LD BC,(6665)
140 DATA 247,82 : ! RST 48,82
150 DATA 247,84 : ! RST 48,84
160 DATA 201 : ! RET
170 !
180 DATA 17,248,25 : ! LD DE,6648
190 DATA 247,211 : ! RST 48,211
200 DATA 237,91,16,26: ! LD DE,(6672)
210 DATA 237,75,9,26 : ! LD BC,(6665)
220 DATA 247,210 : ! RST 48,210
230 DATA 247,212 : ! RST 48,212
240 DATA 201 : ! RET
250 !
260 DATA 237,184 : ! LDDR
270 DATA 201 : ! RET
280 !
290 DATA 237,176 : ! LDIR
300 DATA 201 : ! RET
310 !
320 DATA 125 : ! LD A,H
330 DATA 18 : ! LD (DE),A
340 DATA 19 : ! INC DE
350 DATA 124 : ! LD A,L
360 DATA 18 : ! LD (DE),A
370 DATA 201 : ! RET
400 DELETE 10-
```

1. lista

Parancsszavak

A monitor 5 parancsszót és két ki-
segítő funkciót tartalmaz:

– INS: Hatására a program kérdezi az eltolás értékét, majd a kurzor sorában és az alatta lévő sorban lévő byte-okat az értéknek megfelelően letolja.

– DEL: Ugyanúgy, mint az INS, csak éppen összehúzó.

– LOAD: Hatására a program kérdezi a betöltendő program nevét, hosszát (byte-okban) és azt, hogy hová töltsse (K, cím), majd a program betöltődik.

– SAVE: Értelemszerűen, mint a LOAD (K, cím=honnan). A másolás gombnyomásra indul.

– RUN: A gépi kódú program a „kezdőcímtől” indul.

Kisegítők:

“\$” – ha szöveg elé írjuk, akkor az utána lévő karakterek ASCII szerint letárolódnak.

“j” – az utána következő ugrási címből kiszámolja a relatív eltolás értékét.

Működés

- 170–220: inicializálás + a USRTAB feltöltése
400–570: szerkesztő rész
1000–1090: a parancsszavak figyelése
1110–1140: szubrutin: egy teljes képernyős lista kiírása
1200–1255: INS szubrutin
1300–1360: DEL szubrutin
1400–1510: SAVE szubrutin
1600–1740: LOAD szubrutin
1900–2020: A számok átváltását végző szubrutin
2100–2110: “KESZ” szubrutin
2300–2370: SAVE, LOAD adatbeviteli szubrutin
2400–2480: Hexadecimális átváltó szubrutin
2500–2530: Szöveglehelyező szubrutin
2600–2630: Képernyő felfelé rotáló
2700–2730: Képernyő lefelé rotáló
2900–2910: Bináris átváltó
2920–2950: Relatív ugrás kiszámítás
Fontosabb változók:
KE: kezdőcím
EL: aktuális byte sorszáma
LG: program vége
H: üzemmód mutató

Dörner Péter

A szerkesztő azért van,

hogy a lap olyan legyen,

amilyenek az olvasói!

```

170 GRAPHICS4:DIME**45:CH=68
180 PRINT"TV COMPUTER BASIC MONITOR":PRINT"Copyright 1986 Dörner Péter":PRINT:P
RINTSTR$(FREE):" byte free":PRINT
210 FORI=32042:READE:POKEI,E:NEXT
220 DATA74,26,84,26,29,26,52,26,94,26
400 INPUTPROMPT"Kezdő cím ? ":KE
410 EL=KE:IFPEEK(6764)=0THENLG=E+10:ELSELG=PEEK(6763)*256+PEEK(6762)
415 POKE6764,1:GRAPHICS2
420 IFEL<1E40REL>32767THENSOUND:GOTO400
440 GOSUB1110
470 B$="":PRINTAT12,1,EL:PRINTAT12,10,PEEK(EL):PRINTAT12,16,"? ":
480 A$=INKEY$:PRINTCHR$(CH):CHR$(19):" ":CHR$(19):IFA$="":THEN480
490 IFA$=CHR$(5)THENGOSUB2600:GOTO470
500 IFA$=CHR$(24)ANDL:KE THENGOSUB2700:GOTO470
510 IFA$=CHR$(13)THENGOSUB1000:GOTO470
515 IFA$=CHR$(5)ORA$=CHR$(24)THEN480
520 IFA$="":THENLG=E+10:GOTO400
525 IFA$=" "THENCH=68:H=0:GOTO480
526 IFA$=" "THENCH=72:H=1:GOTO480
527 IFA$=" "THENCH=66:H=2:GOTO480
530 IFORD(A$)=8THEND$=LEN(B$):PRINTA$:GOTO480
535 IFA$<" "THEN480
540 PRINTA$:
550 IFLEN(B$)=45THENSOUND:GOSUB1110:B$="":GOTO470
560 B$=B$&A$
570 GOTO480
1000 '
1010 IFB$="":THENPOKEEL,0:PRINTAT12,11,"0 ":GOSUB2600:RETURN
1015 QW$=CHR$(32ORORD(B$(1)))
1020 IFQW$=" "THENGOSUB1200:GOTO1090
1030 IFQW$="d"THENGOSUB1300:GOTO1090
1040 IFQW$="e"THENGOSUB1400:GOTO1090
1050 IFQW$="l"THENGOSUB1600:GOTO1090
1060 IFQW$="r"THENDPRINTAT12,18," ":I=USR(KE):GOTO1090
1070 IFQW$="$"THENGOSUB2500:GOTO1090
1075 IFQW$="j"THENGOSUB2920:GOSUB1920:GOTO1090
1080 GOSUB1900
1090 B$="":RETURN
1110 CLS
1120 PRINT AT 2,1,:
1130 FORI=EL-10TOEL+11:PRINTI:" ":PEEK(I):NEXT
1140 RETURN
1200 ' INS
1220 INPUTET:IFLG-EL+1<0THENSOUND:RETURN
1240 EXT0,LG,LG+ET,LG-EL+1
1245 FORI=EL TOEL+ET-1:POKEI,0:NEXT
1250 LG=LG+ET:EXT4,LG,6762
1255 GOSUB1110:RETURN
1300 ' DEL
1320 INPUTFH:IFLG-EL+1<0THENSOUND:RETURN
1340 EXT1,EL,EL-FH,LG-EL+1
1345 FORI=LG-FH+1TOLG:POKEI,0:NEXT
1350 LG=LG-FH:EXT4,LG,6762
1360 GOSUB1110:RETURN
1400 ' SAVE
1410 CLS
1420 GOSUB2300
1500 PRINTAT15,1,"Nyomj le egy billentyűt !"
1510 GET:EXT2:GOSUB2100:RETURN
1600 ' LOAD
1610 CLS
1620 GOSUB2300
1670 LG=KE+E-1
1740 EXT3:GOSUB2100:RETURN
1900 ' Átv.
1910 Z=VAL(B$)
1915 IFH=2THENGOSUB2900:IFQ THENQ=0:RETURN
1917 IFH=1THENGOSUB2400:IFQ THENQ=0:RETURN
1920 IFABS(Z)>32767THENSOUND:PRINTCHR$(5):GOTO2020
1940 EXT4,Z,6672
1950 IFABS(Z)>255THENPOKEEL,PEEK(6672):EL=EL+1:POKEEL,PEEK(6673):GOTO1970
1960 EXT4,Z,6672:POKEEL,PEEK(6672)
1970 IFEL>LG THENLG=EL:EXT4,LG,6762
1980 IFABS(Z)>255THENGOSUB1120
1990 PRINTAT12,18," ":PRINTAT12,10,PEEK(EL)
2000 GOSUB2600
2020 RETURN
2100 PRINT:PRINT"KESZ":SOUND$PITCH4000:PRINT:PRINT"Nyomj le egy billentyűt !"
2110 GET:GOSUB1110:RETURN
2300 INPUTPROMPT"File név ? ":C$
2310 IFLEN(C$)>10THENSOUND:GOTO2300
2320 IFLEN(C$)>0THENFORI=6649TO6648+LEN(C$):POKEI,ORD(C$(I-6648)):NEXT
2325 POKE6648,LEN(C$)
2330 INPUTPROMPT"K. cím ? ":KE
2340 INPUTPROMPT"Hossz ? ":E
2350 IFKE<1E40R KE>32767OR E<10R E+KE>32767THENSOUND:GOTO2330
2360 EXT4,KE,6672:EXT4,E,6665
2370 RETURN
2400 ' HEX.
2410 A=LEN(B$):Z=0
2420 FORI=1TOA
2430 IF(B$(I)>"9"ANDB$(I)<"a")ORB$(I)<"0"ORB$(I)>"f"ORLEN(B$)>5THENSOUND:Q=1:RET
URN
2440 NEXT
2450 FORI=A TO1STEP-1
2460 D=ORD(B$(I)):E=A-I
2470 IFD>96THENZ=Z+(D-87)*16^E:ELSED=VAL(CHR$(D)):Z=Z+(D*16^E)
2480 NEXT:RETURN
2500 K=LEN(B$):IFK<2THENSOUND:RETURN
2510 FORI=2TOK:POKEEL,ORD(B$(I)):EL=EL+1:NEXT
2520 IFEL>LG THENLG=EL
2530 EL=EL-K+1:GOSUB1110:RETURN
2600 ' Fel
2610 PRINTAT12,16," ":PRINTAT1,1,CHR$(25):EL=EL+1
2620 PRINTAT23,1,EL+11:PRINTAT23,10,PEEK(EL+11)
2630 RETURN
2700 ' Le
2710 PRINTAT12,16," ":PRINTAT1,1,CHR$(14):EL=EL-1
2720 PRINTAT2,1,EL-10:PRINTAT2,10,PEEK(EL-10)
2730 RETURN
2900 IFLEN(B$)<>8THENSOUND:Q=1:RETURN
2910 Z=0:FORI=0TO7:D=VAL(B$(8-I))*2^I:Z=Z+D:NEXT:RETURN
2920 IFLEN(B$)<2THENSOUND:RETURN:ELSEB$=B$(2:)
2930 I=VAL(B$)-EL-1
2940 IFI<-127OR I>128THENSOUND:RETURN
2950 Z=1:RETURN

```

2. lista

VANNAK MEG CSODÁK

FOLYTATÁS A 17. OLDALRÓL

1987. ÁPRILIS 2.

Telefon a COOPINFORM-ba: – kellene a tíz db nullszériás berendezés. Készen lesz? Igéret, amelyben bízni lehet, de mérget venni rá nem. Első álmatlan éjszakák... Mi lesz ha nem készülnek el a berendezések? Mi lesz ha nem működnek?

1987. ÁPRILIS 14.

Gőzerővel folyik a szervezés. Mind a hét városban keresik azt a gépirónót, aki ismeri az Easy Script szövegszerkesztőt, illetve annak Deltex változatát, keresnek egy olyan számítógépet, aki tud bánni a Commodore-ral, s van kedve részt venni az egészben. És persze keresik a stáb újságíró vezetőjét, meg a szerkesztőség helyét, ahol két telefonvonal is lehet a stábnak. Budapesten folynak a telefonrendelések, a nyomdai előkészítő tárgyalások, az Ötlet fényszedő rendszerének irányítója (aki azonos a BIT-LET Vallató rovatának vezetőjével) tárgyal a tördelővel, a tördelő formátumokat tervez, a grafikus címfejeket rajzol, folyik a stábba szükséges munkatársak szervezése is. Ja, és írják azt a szoftvert, amely alkalmas Easy Script file-ok modemén való továbbítására.

1987. ÁPRILIS 23.

Egy országos értekezleten kiosztásra kerülnek a modemek, a hozzá írt programok, a szövegszerkesztő leme-

zek és a kezelési leírások. Közben igyekszünk a számítástechnikában járatlan újságíróknak is elmagyarázni, hogy ez az egész azért jó, mert a Gyulán gépbe írt anyagot nem kell többször legépelni, s enélkül lesz belőle kinyomott szöveg. Mindez eddig több lépcsőn keresztül volt csak lehetséges. Legévelték az anyagot mondjuk Debrecenben, telefonon vagy telexen fölküldték (telex esetén ez máris még egy gépelést jelentett) Budapesten telefon vagy telexpár alapján legévelték. Ezután megnézte a szerkesztő, javításai alapján újragévelték, majd elküldték a szedőhöz, aki ismét legépelte a szedőgépén.

1987. ÁPRILIS 29.

Főpróba. A helyszínekről megadott menetrend szerint hívják a Commodore Újság szerkesztőségét, ahol örömmel nyugtázzuk, hogy minden a legnagyobb rendben. Az első anyagok a technikai főpróba idején gond nélkül és percek alatt megérkeznek. Azaz, hogy az utolsóként – este kilenckor – jelentkező sárospataki helyszínről semmi sem jön. Ők ugyan látják a képernyőn az üzeneteinket: Helló! Veszítke az adásunkat? De hiába válaszolnak, nálunk csak krikcsok és krakszok jelennek meg a képernyőn. Gyors szerelési tanácsadás a telefonon keresztül, s félórai próbálkozás után sikerül a dolgot megfordítani. Most már hozzánk minden üzenet eljut, de a mi üzeneteink nem jelennek meg a sárospataki képernyőn. Végül is – minthogy Sárospatak messze van – megegyezünk, hogy maradjon így a dolog. Végül is az a lényeg, hogy hozzánk eljussanak a Patakon megírt anyagok!

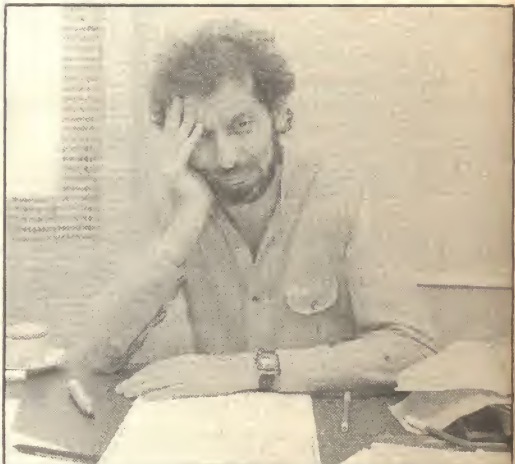
1987. ÁPRILIS 30.

CSÜTÖRTÖK 16.00

A stáb megszállja az Ötlet Budapesten a Jász utcában lévő szerkesztőségét. Fölállítjuk a fogadó állomásokat, amelyek mindegyikére vadonatúj piezós mikrofonokkal szerelt telefonokat tett a posta erre a néhány napra. Ugyanígy átszerelték a vidéki készülékeket is.



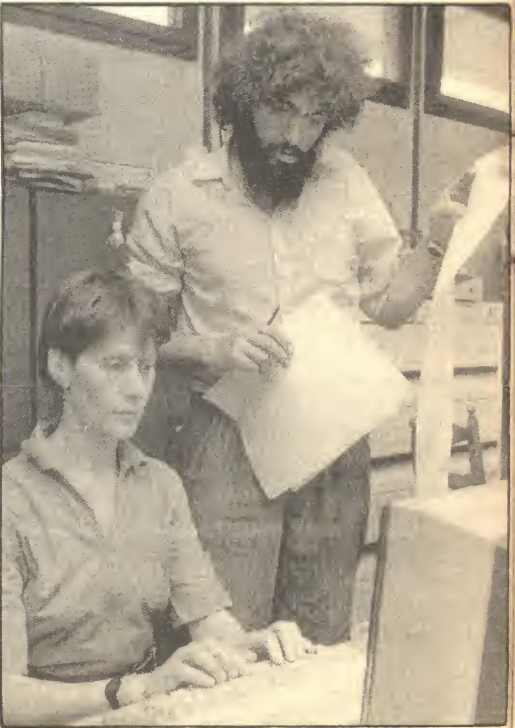
A kész anyagok már hagyományos módon kerültek összerakásra a montírozóasztalon



A szerkesztőnek azért a számítógépes újságinálás közben is akadtak „elkeserítő” pillanatai



Munka közben a fogadó állomások. A kép jobb oldalán jól látható a COOPINFORM akusztikus modemje, amely az első nagy próbán kitűnően viselkedett.

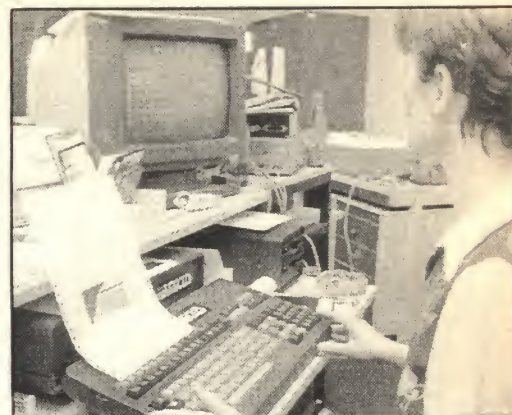


Mindössze egyszer kellett a szöveget legépelni, s a továbbiakban már csak a javítások, kódolások kerültek az Easy Script file-okba

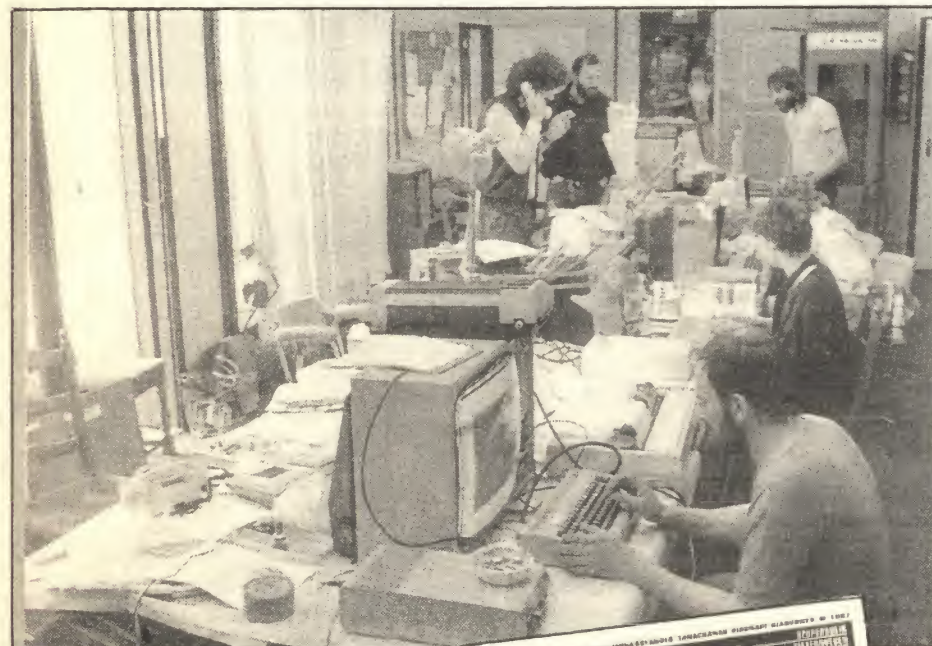
Bár a régebbi szemes mikrofonokkal szerelt telefonokkal is működőképesek a modemek, de mégis ezek a telefonok alkalmasak a jobb minőségű átvitelre, kevesebb a hibalehetőség, márpedig nekünk a gyorsaság most nagyon fontos. A három fogadó állomás mellé települ a három gépirónő, aki profin kezeli a Deltex-et. Van két printelő állás, ahol minden anyag hamar kinyomtatható egy a C64-hez csatolt Epson printeren, amely persze ötször gyorsabb, mint a CBM printerek. A szerkesztőség úgy néz ki, mint egy igazi jövő századi lapszerkesztőség. A másik szobában várja az anyagokat az Ötlet Amicus nevű tördelő computere. Néhány hónapja készült el hozzá az a kiegészítő szoftver, amely lehetővé teszi, hogy az Easy script, illetve a Deltex file-okat egy Plus/4-es segítségével át lehet olvasni a lemezről az Amicusba. Beállítjuk a Plus/4-est is, megérkezik az erre az alkalomra kölcsönzött fénymásológép is, valamint a lapkészítéshez nélkülözhetetlen montírozóasztal. Mindenki elfoglalja a helyét. Kezdődhet a munka.

légát, hogy kezdje újra, mert sok a hiba. Ilyen a három nap alatt 4-5 esetben fordult elő. Az ellenőrzés nagyon lassítaná a munkánkat. Hiszen az Amicusba kerülés előtt úgyis átmegy az anyag a szerkesztők kezén. A hibákat, a szerkesztési változtatásokat ők jelölik a printelt anyagokon. Ezután kerül az a gépirónőkhöz. Ők végzik el a korrigálást. Tulajdonképpen itt van egy hiányzó láncszem a rendszerben. – Ha több időnk lett volna az előkészítésre, akkor a szerkesztő kollégáknak is megtanítottuk volna a szövegszerkesztő kezelését, s akkor egyből ők javíthatnák amit akarnak... Majd legközelebb... A megszerkesztett anyag kerül a tördelőhöz, aki ráírja a szükséges tördelési, tipográfiai utasításokat. Ezután kerül az anyag be az Amicusba, ahol megtörténik a tördelés. Ha ez is kész, mehet az anyag az Amicusból a Monotype Laser comp-jába, ahol a laser fény segítségével fotópapírra kerülnek a megfelelő betűket kialakító pontok. Most már csak az előhívó gépbe kell berakni a Laser comp kazettáját, s kezünkben a

szedés. A következő művelet már csak a kész oldalak összeragasztása, s indulhat az anyag a nyomdába, ahol film, majd lemez készül róla, s végül kinyomják a lapot.



A Commodore-ból a szövegek az Amicus tördelőrendszerbe kerültek. Hogy szedés legyen a szövegből, ehhez még itt is sok kódot, utasítást kellett beírni



Az első elektronikus szerkesztőség így festett madártávlatból

1987. ÁPRILIS 30. – MÁJUS 2.

Lapkészítés számítógépekkel... Gyulán, Egerben, Sárospatakon, Sopronban, Debrecenben, Keszthelyen és Székesfehérváron dolgoznak a diákújságírók. Megírják az anyagaikat, a gépiró beviszi a C64-esbe, majd föltárcsázza valamelyik számunkat, s a lemezen lévő szövegfile-t a modemek és a szoftver segítségével fölküldi. A rendszer kitűnően működik. A posta – igaz mindent előre egyeztetünk – bámulatosan dolgozik. Két helyre – Sárospatakra és Keszthelyre nincs még távhívási lehetőség. Legfeljebb 5 percet kell várni egy kapcsolásra. Az anyagok minimális hibaszázalékkal jönnek föl. A modemekkel maximálisan elégedettek vagyunk. Hogy bízunk benne bizonyítja, hogy a szoftvert szándékosan úgy írtuk meg, hogy ne legyen benne ellenőrzés. A munkatársak ott ülnek a képernyő előtt, s ha úgy látják, hogy 5-10 hibánál több jött egy szövegben, akkor leállítják a vételt, s párbeszéd üzemmódba kapcsolnak. Kéri a vidéken lévő kol-



A végtérkép. Így nézett ki a Tízórai első száma

A stáb Budapesten dolgozó 18 tagja két napra be volt zárva egy épületbe. (Közben kb. három órára azért mindenki elmehetett aludni.) A stáb tagjai nagy izgalommal figyelték, hogy minden rendben halad-e. Valószínűleg mindannyian éreztük, hogy a magyar sajtó és a magyar számítógépes kultúra történetében mindez történelmi pillanat. Hiszen az első teljesen gépesített szerkesztőséget mondhattuk magunknak néhány napig, s ráadásul mindjárt úgy, hogy Budapest kivül hét vidéki városban is működtek a szerkesztőségeink. A technika kitűnően vizsgázott. Egyetlen gép, egyetlen modem sem romlott el. Az Amicus, a Laser comp is jól bírta, bár a gyári előírások szerinti 18 Celsiusra temperált szoba helyett a májusi 30 fokban kellett dolgozni. A cím nem véletlen. Valóban a csodával határos számunkra, hogy a dolgok működésképtelenségéről híres kelet európai földrészen ilyen zökkenőmentesen sikerült ez a vállalkozás. Még a posta legújabb, mindössze kísérleti stádiumban lévő szolgáltatása a konferenciakapcsolás is összejött. Első este ugyan – amikor szerkesztőségi megbeszélés céljából köttettük össze a hét helyszínt és magunkat – még voltak hibák. Eger és Debrecen kimaradt a vonalból. De a következő két napon amikor a rendezvényen részt vevő diákokat hoztuk össze egy dumapartira, hogy cikket írjunk belőle, már tökéletes volt a kapcsolat.

Tűz volt az állítani, hogy meghatódunk. De az biztos, hogy mindannyian sajnáltuk amikor befejeződött a Tízórai akció. A szerény nyolcezer példányban megjelent újságocská ugyanis ezt a címet viselte: Tízórai. Az akcióban részt vevők számítógépesek és újságírók egyaránt úgy éreztük, hogy valami olyasmit csináltunk, amire hivatászerűen nem lesz módunk egy darabig. Mert mindez – a számítógépes újságcsinálás – csak egy kísérlet volt. Sikerült – ami egy kicsit csoda, de hogy meghonosodna, napi munkamódszerré válna a következő tíz esztendőben – hát ebben bizony egyikünk sem reménykedik. Különösen nem a BIT-LET szerkesztője:

Angyalosi László

800XL ATARI KÖR



PROGRAMAJÁNLAT – PLAYER-MISSILE GRAFIKA

Az ATARI számítógép tud sprite-okat kezelni, bár erről a géphez adott többnyelvű füzetek nem szólnak.

E grafikai funkció lényege, hogy bizonyos általunk definiálható alakzatok a képernyőn gyorsan mozgathatók (a koordinátáik megadásával), a kép eredeti állapotának megváltoztatása nélkül (=szellemgrafika). Ez a mozgatás nem szoftver úton (vagyis a képernyőadatokat módosíthatásával), hanem hardver úton történik: a memóriában különböző helyen levő képernyőmemóriát és a sprite-táblázatot a videoprocesszor megfelelő módon összevegyíti.

Az ATARI 800XL kétféle sprite-ot ismer:

4 db egyenként nyolc képpont széles (Player).

4 db egyenként két képpont széles (Missile).

Érdekességük, hogy mind egy-egy 256 sor magas sáv, ezzel a képernyőt teljes magasságban kitöltik.

Az alábbi kezelést megvalósító memóriacímek POKE utasítással használhatók:

53248–53251 a négy Player vízszintes pozíciója.

53252–53255 a négy Missile vízszintes pozíciója. A 0, ill. 255 kívül esik a látható képernyőn (bal oldalon a 0), így a sprite-ok teljesen „levihetők” a képről.

A hardver a sprite-ok függőleges mozgatására nem ad lehetőséget.

53256–53259 a Player-ek szélessége:

0, ill. 2 a normál szélesség (egyszeres).

1 a dupla szélesség.

3 a négyszeres szélesség. Így a teljes Player „megvastagítható”, ill. „vékonyítható”.

53260 ugyanez a Missile-okra, de itt bitpárok vezérelnek, tehát a négy Missile független vezérlését a következő módon oldhatjuk meg (itt az S0...S3 a szélességet tartalmazó változók):

POKE 53260,S3*64+S2*16+S1*4+S0

704–707 a sprite-ok színregiszterei: az azonos sorszámú Player és Missile azonos színű. A színt a következő módon állíthatjuk be:

POKE 704+X,16*szín+fénység

(X a sprite száma: 0–3)

623 prioritás-regiszter: a sprite-ok a képernyőábrákhoz viszonyított „térbeli” helyzetét szabályozza: vagyis azt, hogy a sprite egy ábra előtt vagy mögött legyen.

Jegyezzük meg a sprite-ok egymás közti helyzetét, melyet a prioritás-regiszter nem befolyásol:

legelől áll a 0. Player, leghátul a 3. Missile.

A prioritáskódokkal érdemes kísérletezni, itt csak három emelet ki:

1 – minden sprite az ábrák előtt.

4 – mindegyik az ábrák mögött.

8 – mind a 0. és 1. színregiszter mögött (SETCOLOR), 2–3. színregiszter előtt.

5 REM ***** SPRITE ATARI *****

6 REM

10 POKE 106,144:GRAPHICS 0

20 FOR I=38912 TO 39283:READ A:POKE I,

A:NEXT I

30 FOR I=39360 TO 39679:POKE I,0:NEXT I

40 FOR I=0 TO 7:A=USR(39133,0,I):NEXT I

50 NEW

100 DATA 41,7,141,117,153,170,189,118,

153,141,142,153,189,126,153,141,143,

153,189,134,153

110 DATA 141,144,153,138,201,4,41,3,

141,145,153,170,169,153,141,204,0,169,

192,141,203,0,169,0,141

120 DATA 212,0,169,155,141,213,0,144,

14,169,128,42,42,202,16,251,56,141,146,

153,160,0,96,138,106,106,106

130 DATA 141,203,0,238,204,0,238,213,0,

202,16,250,138,24,144,230,104,240,63,

201,3,208,63,104,104

140 DATA 141,147,153,104,104,73,255,41,

63,141,148,153,104,104,32,0,152,144,17,

174,145,153,173,147,153

150 DATA 41,3,44,10,10,202,16,251,141,

147,153,173,146,153,73,255,172,148,153,

49,203,13,147,153

160 DATA 145,203,160,0,76,30,153,104,

240,77,201,3,208,67,104,104,41,15,141,

147,153,104,104,10,10,10

170 DATA 10,13,147,153,170,104,104,41,

3,168,138,153,192,2,76,234,152,104,240,

41,201,2,208,31,104

180 DATA 104,233,1,41,3,168,104,104,72,

41,7,170,152,157,118,153,104,32,0,152,

76,30,153,104

190 DATA 240,10,201,2,240,22,170,104,

104,202,208,251,169,152,141,7,212,169,

3,141,29,208,169

200 DATA 62,141,47,2,96,104,141,148,

153,104,141,147,153,104,104,32,0,152,

174,117,153,173,147,153

210 DATA 141,143,153,157,126,153,173,

148,153,41,1,141,144,153,157,134,153,

32,101,153,136,140,149,153

220 DATA 160,63,173,144,153,74,170,173,

143,153,176,10,240,183,141,149,153,206,

149,153,144,3,141

230 DATA 212,0,173,142,153,141,150,153,

238,149,153,208,17,206,149,153,177,203,

45,146,153,1,212

240 DATA 129,212,238,212,0,240,144,206,

150,153,16,229,136,16,220,76,234,152,

173,146,153,73

250 DATA 255,170,49,212,145,212,138,

200,208,248,96

Az általam készített program a következő plusz lehetőségeket adja (utasítás USR függvényével pl. A=USR...):
USR(39001,kód,pozíció,sorszám)

– a sprite-ok magasságát 64 sorra korlátoztam, így a kódot a 0–63 sorba (pozícióba) lehet elhelyezni. A kód értéke Player esetén 0–255, Missile esetén 0–3 közti lehet. A sorszám 0–3 a Player-eket, 4–7 a Missile-okat választja ki. A kód az ún. bitmintából adódik. Pl. legyen a Player egyik sora a következő alakú:

..*.*.*.*

A "*" jel az adott sort ábrázoló pont. Ezt alakítsuk át bináris számmá oly módon, hogy ezen pontok értéke legyen az 1!

00101101

Váltsuk át decimális számmá: a jobb oldali számjegy helyiértéke 1, az összes többi jegy a tőle jobbra levő kétszerezését éri.

$32+8+4+1=45$

Az USR függvénybe ezt a kódot írva a fentebbi ábra jelenik meg (persze csak, ha az adott sprite megfelelően pozícionálva van, nem takarja semmi stb.)

USR(39066,fényesség,szín,sorszám)

– A 704–707 sprite-színregiszterek beállítását végzi, lényegében a POKE utasítással egyenértékű.

A sorszám itt 0–3 közti lehet.

USR(39102,méret,sorszám)

– Ezzel a rutinnal a sprite-ok függőlegesen nagyíthatók: az 1–4 méretszámok (itt nem 0–3!!) az egyszerűsítéstől a négyszeresig növelik a sprite magasságát. Így módon a 64 soros sprite négyszeresre nagyítva betöltheti a teljes képernyőmagasságot.

USR(39133,pozíció,sorszám)

– A sprite-okat függőlegesen mozgatja. A sprite 0. (legfelső) sora kerül a megadott pozícióra: a képernyő teteje felett van a 256-os, az alja alatt az 511-es pozíció. Ez azt jelenti, hogy a függőlegesen négyszeresre nagyított sprite (=256 sornyi magas) is mind felfelé, mind lefelé teljesen levihető a képről (0. ill. 511 pozíció).

A program működéséről: a lefuttatás után törli magát, ettől kezdve a RESET gomb megnyomásáig a négy USR rutin működik (a RESET után azonnal kiadva a POKE 106,144:GRAPHICS0 parancsot, a rutinokat, sőt sprite-jainkat is megmenthetjük, csak a pozíciójukat kell újra beállítani.)

– Az USR rutinok a felsorolt funkciókon kívül a sprite-ok működéséhez szükséges vezérlő-regisztereket is kezelik a videoprocesszorban (ezek nélkül a sprite-ok nem jelennek meg a képernyőn). Mivel azonban a GRAPHICS utasítás ezeket a vezérlőregisztereket átállítja, utána célszerű mindig kiadni valamelyik USR függvényt a fentiek közül – alkalmasint adatok nélkül, A=USR(cím) formában. Ekkor ugyanis nem kapunk hibajelzést; az adott funkció hatástalan marad, de a vezérlő-regiszterek értékeit a rutin beállítja. Ha ezt nem tesszük meg, a sprite-ok zavaró villódzás formájában jelennek meg a képernyőn.

– A BASIC betöltő program a 10. sorban lejjebb helyezi a képernyőt a memóriában, majd az így felszabadult helyre a 20. sorban betölti a gépi kódú programot. A 30. sor törli a sprite-területet, a 40. sor pedig alapállásba állítja a sprite-okat.

Mivel az 50. sor törli a BASIC programot, az első lefuttatás előtt feltétlenül mentsük szalagra vagy lemezre!

– A DATA sorok hibátlan begépeléséről az alábbi parancssorral győződhetünk meg:

RESTORE :A=0:FOR I=1 TO 372:READ B:A=A+B:
NEXT I: A

A képernyőn a 46882 számnak kell megjelennie.

A gépi kódú rutinok számára 4 Kbyte helyet foglalunk le. Ebből ugyan most csak 2 Kbyte-ot használunk (a képernyőmemóriának fontos, hogy kerek 4 Kbyte-os címen legyen vége), a fennmaradó területet remélhetőleg a jövőben még betöltjük rendszerbővítő rutinokkal.

Rieth József

APRÓSÁGOK

Tisztelt Szerkesztőség!

Második osztályos gimnáziumi tanuló vagyok. Négy éve programozok. Eddig Commodore-on, jelenleg ATARI 800 XL-esen. Nyáron kaptam szüleimtől Datasettel együtt. Magyar nyelvű leírásom mindössze az „Atari hetedhét” című könyv. Ettől többre voltam kíváncsi, ezért sokat „fag-gattam” a gépet.

Amire rájöttem, szeretném közreadni.

Az alábbi címek szerepe:

755,4 – a képernyő karaktereit „fejre” állítja

755,2 – visszaállítja a karaktereket

82,x – a képernyő sorban a bal margó hosszát állítja be (max. 39)

83,x – a képernyő sorának utolsó pozíciója (jobb margó)

767,1 – a BREAK és RESET billentyűn kívül minden billentyűt letilt

767,0 – engedélyezi a billentyűket

53817 értéke 17 – ha a HELP billentyű le van nyomva

53279 értéke attól függ, melyik „fém” billentyű van lenyomva.

értéke: 7 – ha egyik sem

6 – ha START

5 – ha SELECT

4 – ha SELECT+START

3 – ha OPTION

2 – ha OPTION+START

1 – ha OPTION+SELECT

0 – ha OPTION+SELECT+START billen-tyű van lenyomva.

40000–40960 – képernyő memória helye.

52232–53248 karaktermemória helye.

Nyikes Gábor, II. oszt. gimn. tanuló, 3400 Mező-kövesd, Vilma út 62.

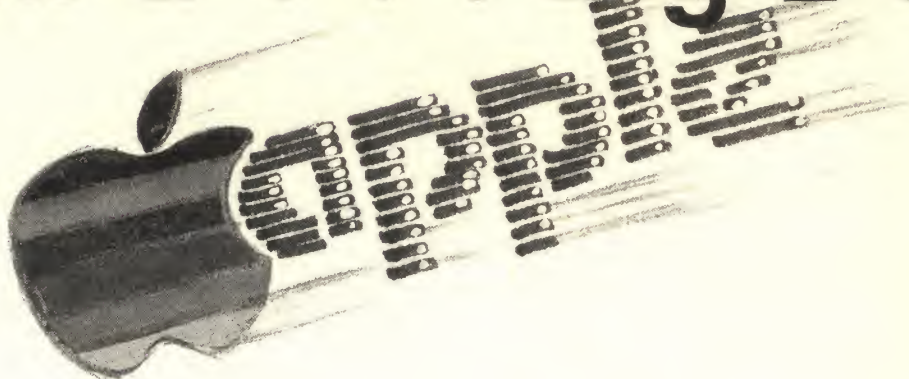
KERAVILL MEV
ELEKTRONIKAI
MÁRKABOLT

BP. V., MŰZEUM Krt. 11.

MIKROELEKTRONIKA:
A JÖVŐ A JELENBEN.

FÉLVEZETŐK,
INTEGRÁLT ÁRAMKÖRÖK,
MIKROPROCESSZOROK
ÉS CSATLAKOZÓK.
SZAKTANÁCSADÁS, CSOMAGKÜLDŐ SZOLGÁLAT.

Almavirágzás?



Még el sem ültek az Amiga gépek megjelenését és az 1000-es csöndes bukását követő hullámok, ismét új gép tűnt fel a láthatáron: az Apple (alma) cég „asztali óriása”, a Macintosh II. Az intenzív piaci rohamba fogott 3A trió (Atari-Amstrad-Apple) oszlopos tagja ismét visszakerült megszokott helyére a nyugati szaksajtó címlapjára. Mivel azonban az Apple gépek náunk nem olyan közismertek, mint a Sinclair vagy Commodore típusok, tekintsük 3 rövid fejezetben át a cég eddigi tevékenységét!

1. FEJEZET – AZ INDULÓ ÉVEK

Az első igazi nagy sikerről az Apple II-ről mindenki tudja az alapsztorit. Ez az a gép, amelyet egy alagsori garázsban tervezett két fiatal, Steve Wozniak és Alan Baum 1976-ban. Az addigi gépeknél jobb, színes grafikája, bővítési lehetőségei, perifériái és remek programkínálata az akkori félprofesszionális kategória élére emelték. A fejlesztők természetesen a végsőig igyekeztek kiaknázni a gépben rejlő lehetőségeket, gyors ütemben piacra dobták az Apple II+-t, majd a IIe-t, később a IIc-t. Az utóbbi típusból az árcsökkenés érdekében kihagyták a bővítési lehetőségeket, valószínűleg ez volt az oka, hogy a IIc végül megbukott. Az 1984-es gazdasági összességében döbbenetes eredményt hozott. A bevételek legnagyobb része az akkor már számos újabb gyártmány mellett még mindig piacon levő Apple IIe eladásából származott. Így tehát a cég hálál gesztusként kitűzte a lobogót: Az Apple II. örök!

1985-ben nagy erővel hozzáláttak az Apple II. korszerűsítéséhez. Az új gép Apple IIGS néven jelent meg. A 8 bites 6502 processzort felváltotta a 650816, amely 16 bites, de működhet 6502 üzemmódban is. 7 db bővítő és egy SCSI csatlakozóval (erre később részletesen kitérek) látták el. A képernyő legnagyobb felbontása 640*200, ekkor soronként 4 szín jelölhető ki egy 4096 színből álló palettáról. Beépített floppy nincs – az ősz Apple II-höz hasonlóan – viszont könnyen illeszthető 3,5 coll és 5 1/4 coll átmérőjű meghajtó. A beépített memória 256 Kbyte méretű, bővítés 8 Mbyte-ig lehetséges. A gép fő erőssége mégsem a hardver, hanem az a becslések szerint 25 000 program, ami a GS megjelenésekor már piacon volt.

2. FEJEZET – A NAGY ÚJÍTÁSOK

Időközben új szelek kezdtek fújdolgalni a számítógépi piacon. Megjelent a „nagy kék”, az IBM. Talán ez volt az oka, hogy az új reménység az Apple III. példátlanul rövid idő alatt megbukott, de valószínűbb, hogy az Apple maga okozta gyártmánya halálát Macintosh fantáziánévű termékével. Az új gép az Apple fejlesztési stílusára jellemzően nem egyszerűen nagyobb volt elődeinél, hanem mi-

nőségileg hozott újat. Az első változat 128 Kbyte memóriával készült, ezt később 512 Kbyte-ra bővítették. Merész újítás volt a mindössze 9 inch képátlójú fekete-fehér monitor, hisz az IBM gépekhez 12 inches színes monitorok is illeszkedtek. Hogy ez miért volt igazán újítás? Mert a kis képernyőn 512*342 képpont felbontást valósítottak meg, ami pazar látványt nyújt.

A szokatlan húzások között a második – és valószínűleg a lényegesebb – az azóta elmaradhatatlan egér. A hajlékony kábelrel csatlakoztatott dobozkát az asztalon tologatva lehet a képernyőn levő mutatót mozgatni és segítségével utasításokat, üzemmódokat választani, vagy akár rajzolni. Az egér valójában nem egyszerűen egy új beviteli eszköz volt, hanem az új programozási stílus része. Az Apple tulajdonképpen ezt a stílust már a Mac-et megelőzően kipróbálta nem túl nagy karriert befutó Lysa nevű gépével. A fejlesztőkben felmerült a gondolat, hogy az operációs rendszerekhez mellékelt arasznyi vastagságú dokumentáció megtakarítható, ha a gép egy kicsivel okosabb. Elhagytak hát egy csomó összetett, ám igen ritkán használt parancsot, az elérhető parancsok pedig egy menün jelennek meg, amelyről az egérrel lehet választani. Így pl. míg egy program másolása az IBM PC-ken futó MS-DOS-ban a következő utasítással megy: copy a:programnév b: addig az új ún. ikonvezérlésű operációs rendszerben a felhasználó rámutat a program nevére, majd a meghajtott jelképező fiókra és a másolás már megy is. Nem csoda, hogy a Macintosh az IBM PC-k komoly riválisává vált.

A sikerhez hozzájárult a Macintoshban alkalmazott Motorola 68000 processzor is. Az elsősorban ipari és grafikai célokra széles körben elterjedt processzorcsaládot sokan a jövő nagy reménységének tartják. A 68000-es processzor képességeiben már sokkal inkább emlékeztet a nagy gépek központi egységeire mint mikroprocesszorra. Adatvezetéke 16 bites, belső regiszterkészlete 8 db 32 bites címregiszterből és 8 db 32 bites adatregiszterből áll. A címregiszterek egyike a stackmutató. A regiszterek egyenértékűek, valamennyi utasítás valamennyi regiszterrel végrehajtható, a címzési módok konzekvensek, a 14 fajta címzési mód mindegyike minden helyzetben alkalmazható. Az operációs rendszer és a felhasználói programok elkülönítése érdekében külön felhasználói és rendszerállapot létezik. A közvetlenül elérhető címtartomány 16 Mbyte, ehhez 24 címvezeték áll rendelkezésre.

Összehasonlításképpen: az IBM-XT gépekben levő Intel 8088 adatvezetéke 8 bites, regiszterkészlete 4 db 16 bites regiszterből, 4 db mutatóregiszterből és 4 db szegmensregiszterből áll. Ez utóbbiakra a 64 Kbyte-os memóriaszegmensek eléréséhez van szükség. Ez persze lényegesen körülményesebb memóriahozzáférést eredményez. Az uta-

sításkészlet akkumulátor orientált, a regiszterkészlet nem egyenértékű regiszterekből áll.

Az Apple-től megszokott módon a Macintosh sikerét igyekeztek ismét jól kihasználni. Piacra került a Macintosh Plus, vagy ahogy rajongói át keresztelték, a Mac Plus. A memóriát 1 Mbyte-ra bővítették, a 400 Kbyte-os 3,5 collos floppy kapacitását 800 K-ra növelték és a rendszer ROM is gyarapodott: 64 Kbyte-ról 128 Kbyte-ra. Ebben egyébként már nem kapott helyet a népszerű Quick Draw rajzoló program, ehelyett megjelentek a winchesterkezelő rutinok és megint valami új, az SCSI busz.

A buszt eredetileg lemezegységek illesztésére szánták, később kiderült, hogy számítógép-számítógép kapcsolat lebonyolítására is alkalmas. (Innen a név: Small Computer System Interface.)

A buszra 8 egység kapcsolódhat, ezek mindegyike lehet floppy, keménylemez, számítógép vagy bármilyen adatfeldolgozó eszköz. Az információforgalmat bármelyik egység kezdeményezheti, maga az információcsere pedig igen magas szinten történik. A kapcsolat kezdeményező eszköznek a megszólított eszköz mibenlétével nem kell foglalkoznia. A megszólított eszköz kérheti a kapcsolat felfüggesztését, majd ha képes az igények kielégítésére, újra följújthatja a kapcsolatot. Így a rendszer buszkihasználtsága igen jó, maga a hálózat pedig a kezelőprogramok átrfása nélkül egyszerű „dugdosással” átkonfigurálható.

A lendületet ismét kudarc törte meg. A következő fejlesztés a Mac SE nem váltotta be a tervezői reményeket. Ez nem is meglepő, mivel lényegi újítást nem tartalmazott.

3. FEJEZET – ÉS A JÖVŐ?

1987 áprilisában megjelent a Macintosh II. Központi egysége a 68020 szuperprocesszor, a 68000 alaposan továbbfejlesztett változata. Ez már igazi 32 bites processzor saját belső gyorsítótárral (cache memory), melybe nemcsak az aktuális utasítást veszi fel feldolgozásra, hanem annak környezetét is. Így akár egész ciklusok futtathatók anélkül, hogy utasításért a memóriához kéne fordulni.

A 68020 kiegészíthető 68881 aritmetikai segédprocesszorral. A Mac II-be behelyezve ezt az IC-t az aritmetikai műveleteket végző programok 3...30-szor gyorsabban futnak. A cég szerint, ha a programozó kifejezetten épít a 68881-re, a sebesség 30...2000-szeresére is növelhető.

Az IBM-AT méretű dobozban 6 db ún. NuBus csatlakozó is helyet kapott. Ezek mindegyikébe akár processzorkártya is helyezhető és így a 68000 család multiprocesszoros támogatásának köszönhetően valódi többprocesszoros rendszer jön létre. Így kívánják egyébként a közeljövőben megjelenő Intel 80286 segédprocesszort is csatlakoztatni, minek eredményeként a felhasználó asztalán egy dobozban egy Macintosh II és egy IBM-AT kaphat helyet. Ráadásul a két gép párhuzamosan működhet.



Almavirágzás?



Hogy a dolog simán menjen, a Mac II-höz két monitor is csatlakoztatható. A grafikai felbontás 640*480, minden képponthez 16 szín választható egy 16 millió színből álló palettáról.

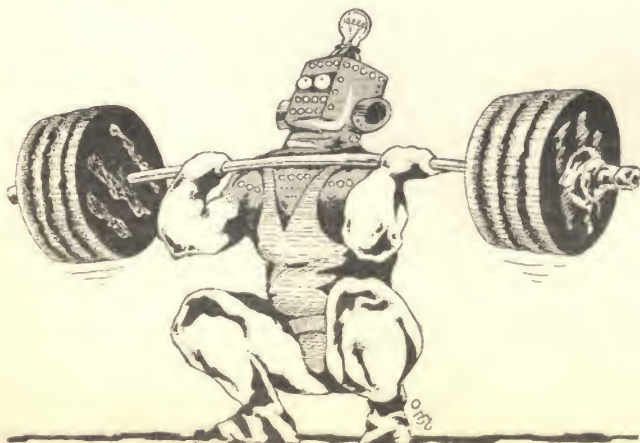
A 256 Kbyte-os video RAM-ot 512 Kbyte-ra bővítve a megadható színek száma 256. A második monitor csatlakoztatását a gép észreveszi és egyéb beavatkozás nélkül alkalmazkodik. A rendelkezésre álló tárhelykapacitás 1 Mbyte, de ez 2 Gbyte-ig bővíthető. A 3,5 collos meghajtó 800 K-s, lehetőség van további 2 floppy és egy 20, 40 vagy 80 Mbyte-os keménylemez beépítésére. Az SCSI buszon keresztül további floppy-k és keménylemezek csatlakoztathatók.

A hangot egy négycsatornás szintetizátor és két Sony erősítő szolgáltatja – természetesen sztereóban. A szintetizátor a hagyományos hullámformákon kívül egy szokatlan szolgáltatást is nyújt: képes mintavételezett hangok visszaadására. A mintavételezés 44,1 KHz-en történik – csakúgy, mint a CD játszóknál – a kívánt hullámforma pedig 256, illetve 512 amplitúdóadattal adható meg. Másik lehetőség külső jel mintavételezése, ezt a szintetizátor a tárolás után tetszőleges hangfekvésben képes visszaadni. Így klavíratúra csatlakoztatás után (mert természetesen azt is lehet) az ember eljátszhatja kedvenc dallamait mondjuk sörözüvegben, egy személtapát és hajnali mozdonyfütyty kíséretével.

A hardver lehetőségeket jól összehangolt ikonos vezérlésű operációs rendszer fogja össze, mely természetesen az egérre támaszkodik. A cég óvatosságot mutat, bizonyítja, hogy a merész újítások mellett is fut a gépen a régi Mac-programok 95%-a, a 80286 kártya megjelenése után pedig a hatalmas MS-DOS programválaszték is rendelkezésre áll. Tervezik az UNIX operációs rendszer teljes adaptációját is. Egy változat ugyan már működik a gépen, de a végleges verzió csak a 68851 MMU (memória-vezérlő) megjelenése után várható.

Hogy mi lesz az új gép sorsa, az egyelőre rejtély. A magas, közel 4000 dolláros ár minden esetre izgalmassá teszi a kérdést. A piac úgy tűnik bizalmat szavazott az Apple-nek, a Wall Street-i részvények árának gyors emelkedése legalábbis erre vall. Az Amiga 1000 bukása fenyegető példa, de talán a Mac II. lesz az a gép, ami megtöri az IBM-gépek és klónjaik uralmát, előtérbe hozva a felhasználó személyét az operációs rendszerek mágikus parancsszóáradatával szemben.

FAZEKAS ISTVÁN (A Byte nyomán...)



PROGRAMA.ÍRÁNLAT:

C 16 KARAKTER EDITOR

C 16 KARAKTER EDITOR

Régi vágyam volt, hogy legyen egy saját készítésű karaktereditorom. Most végre elkészült, és rövidsége miatt szeretném másokkal is megismertetni.

A PROGRAM RÖVID ISMERTETÉSE:

Az első öt sor végzi a gépi kódú átmásolórutin betöltését, indítását és az interpreter átállítását. A 70–90 sor végzi az átírandó karakter beolvasását. A 100-as sor határozza meg a karakter kódszámát, valamint első byte-jának helyét a memóriában. A karakter mind a 8 byte-ját a 120-as sor olvassa be. A 130–150 sor a 8 byte decimális alakját váltja át binárisra, és írja ki a képernyőre. A 170-es sor CLR utasítására azért van szükség, mert a program a grafikus karakterek átirásakor néha „megzavarodik”. Az új karaktert – binárisan ábrázolva – a 250-es sor kéri be. Ennek korrektségét ellenőrzi a 260–280 sor. Az új karakter tárolását a 310–330 sor végzi.

A PROGRAM VÁLTOZÓI:

W: számláló, E: az átírandó karakter kódja, C: az átírandó karakter első byte-jának helye a memóriában, A\$(x): a karakter byte-jai ábrázolva, Y\$(x): az új karakter binárisan ábrázolva, M,X: segédváltozók, az új karakter memóriába helyezésénél van szerepük.

HOGYAN HASZNÁLJUK A PROGRAMOT?

RUN után a program megkérdezi, hogy melyik karaktert írja át. Válaszoljunk a megfelelő billentyűvel. Ezután ki-rajzolódik a képernyőre az adott karakter nagyított képe.

```
10 PRINT "C": W=0
20 FOR A=0 TO 32: READ B: POKE 12288+A, B: NEXT
30 DATA 160, 8, 162, 8, 129, 8, 208, 157, 8, 56, 232
35 DATA 208, 247, 238, 6, 48, 238, 9, 45, 136, 208, 236
40 DATA 169, 208, 141, 6, 48, 169, 56, 141, 9, 48, 96
50 SYS 12288: POKE 65298, 192: POKE 65299, 56
60 COLOR 0, 16, 0: COLOR 4, 15, 3: COLOR 1, 11, 6
70 PRINT: PRINT "MELYIK KARAKTERT ÍRJAM AT?"
80 GETKEY A$: S=INSTR(1, A$: "A-Z")
90 CHAR, 1, 1, "": CHAR, 2, 1, A$: CHAR, 3, 1, " "
95 E=PEEK(3114): C=14336+8*S
100 E=PEEK(3114): C=14336+8*S
110 PRINT CHR$(13): RESTORE 180
120 FOR A=0 TO 7: S(A)=PEEK(C+A): NEXT A: DO
130 B=S(W): FOR F=0 TO 7: READ Q
140 A(F)=B/Q: IF A(F)=1 THEN A(F)="1": B=B-Q: ELSE A(F)="0"
150 NEXT: PRINT CHR$(27)+"Q": RESTORE 180
155 FOR A=0 TO 7: CHAR, A+13, W+13, A(A): NEXT: W=W+1
160 LOOP UNTIL W=8
170 CLR: E=PEEK(3114): C=14336+8*E: W=0
180 DATA 128, 64, 32, 16, 8, 4, 2, 1
190 CHAR, 13, 5, "AZ ÚJ KARAKTERT SORON-"
200 CHAR, 13, 6, "KENT IRO BE!"
210 CHAR, 13, 8, "URES PONT="0"
220 CHAR, 13, 9, "TELI PONT="1"
230 CHAR, 0, 13, " "
240 FOR A=0 TO 7
250 CHAR, 11, 13+A, "": INPUT Y$(A)
260 IF LEN(Y$(A)) < 8 THEN 250
270 FOR Z=1 TO 8
280 IF MID$(Y$(A), Z, 1) < "1" AND MID$(Y$(A), Z, 1) < "0" THEN 250
290 NEXT: RESTORE 180: NEXT: W=0
300 E=PEEK(3114): C=14336+8*E: DO
310 RESTORE 180: FOR A=1 TO 8: READ Q
320 M=VAL(MID$(Y$(A), A, 1)): X=X+(M*Q)
330 NEXT: POKE C+W, X: W=W+1: X=0: LOOP UNTIL W=8: W=0: GO TO 70
```

0 jelöli az üres, 1 a tele képpontokat. A karaktert soronként írhatjuk át. Az előzőleg beírt sort nem tudjuk javítani, csak ha végigírjuk mind a nyolc sort, és újra kezdjük. A karakterek átirását addig folytathatjuk, míg a „melyik karaktert írjam át” kérdésre RETURN-nal válaszolunk. Ha el akarjuk menteni az új karakterkészletet, kapcsoljunk át monitorba és adjuk ki a következő parancsot:

S"FILENEV",01,3800,4000

(Drive esetén a 01 helyett 08-at írjunk!)

Megjegyzés: ha az új karaktereket akarjuk használni, ne kapcsoljunk be a finomgrafikát. Ha használatukkor bármilyen – a gép által jelzett hibát – követünk el, a képernyő zavarossá válik. Ilyenkor a POKE 65298, 192: POKE 65299, 56 utasítás segít. Ezt érdemes KEY-be beírni.

A program a rövidség érdekében nem tartalmazza az újonnan bevitt karakter ellenőrzését, de ezt bárki elkészítheti. A program használatához mindenkinek sok sikert kívánok.

Eöry Zsolt, 2626 Nagymaros, Rákóczi út 30.



VALÓSZÍNŰSÉGSZÁMÍTÁS

Rendhagyó módon egy picit elmélettel kell kezdeni, minthogy nem egyértelműen törzsanyagról van szó.

– A valószínűség szóról többé-kevésbé mindenkinek van valamilyen fogalma. Sokan egy százalékos rendeltetéshez (pl. első valószínűsége 35%, pénzzel fejte dobunk, ennek az esélye 50% stb.) Korrektebb a valószínűséget egy nulla és egy közé első számként értelmezni: az esetek hányad részében következik be a vizsgált esemény (dolog).

– Véletlen eseménynek egy olyan történetet nevezünk, aminek pontos lefolyása, vagy következményei többféleképpen lehetnek. Azt általában tudjuk, milyen lehetőségek vannak, de hogy pontosan melyik lesz, arra vonatkozóan csak találgatni lehet. Például egy kocka feldobásakor az 1–6 számok bármelyike lehet legfelül, de hogy melyik, az teljesen bizonytalan.

– A véletlen események lehetséges kimenetelét azok a lehetőségek, amelyek közül valamelyik bekövetkezik. Kocka dobásnál az 1,2,3,4,5,6 számhalmaz; időjárásnál a (napos, borult, esős, szeles, változó) halmaz egy lehetséges megadás bár ehhez nyugodtan hozzávehető a „kicsit felhős”, és az „ítéletidő” is. Egyértelműség kedvéért kizárólag olyan eseteket tárgyalunk, ahol a kimenetelnek számokkal jellemezhető. Például nem azt mondjuk, hogy „semmi-kevés-sok-rengeteg” csapadék lesz, hanem inkább azt, hogy X mm – egyszerűség kedvéért egészekre kerekítve.

– Egyenletes eloszlású (valószínűségű) eseményről akkor beszélünk, ha valamilyen lehetséges kimenetel egyforma valószínűségű (esélyű). Például a kocka dobásnál az 1–6 értékek egyformán 1/6 eséllyel adódnak, de nevetséges az az állítás, hogy 1/2 valószínűséggel esik a hó április végén.

– A valószínűségek „kiszámítása” két-féleképpen történhet: definíció szerint az összes lehetőség számával el kell osztani az ún. kedvező esetek számát (például a kockával azért dobunk 1/2 valószínűséggel párosat, mert a hat lehetőségből a fele páros, a fele páratlan, tehát $3/6 = 1/2$ a kívánt arány). A második út, hogy gondolatban (avagy pld. computerrel) sokszor megismételjük a véletlen eseményt, és a kedvező/összes esetek arányát tekintjük valószínűségnek. Ezt az utóbbit módszert szimulációnak nevezik: úgy teszünk, mintha (sokszor) megtörténne. Itt nagyon fontos a sokszori „eljátszás”, a szimuláció megbízhatósága elsősorban ettől függ.

Maguk az eljárások konkrét esetekben sokkal bonyolultabbak, több elemzett példával próbáljuk pontosítani a módszereket.

1. feladat: „Dobjunk fel” számítógéppel egy kockát 1000 alkalommal, és értékeljük ki a kísérletet. Ezt különben a kockadobás szimulációjának nevezzük.

```
50 N=1000
100 FOR I=1 TO N
110 A=INT(RND(1)*6+1):KO(A)=KO(A)+1
120 NEXT I
200 FOR I=1 TO 6:PRINT KO(I);KO(I)/N;KO(I)/N-1/6:NEXT I
```

2. feladat: Szimuláljunk két kockával való dobássorozatot!

```
20 DIM KO(20)
50 N=1000
100 FOR I=1 TO N
110 A=INT(RND(1)*6+1)+INT(RND(1)*6+1):KO(A)=KO(A)+1
120 NEXT I
200 FOR I=2 TO 12:PRINT KO(I);KO(I)/N:NEXT I
```

Hangsúlyozzuk, hogy a $2*INT(RND(1)*6)+1$ és az $INT(RND(1)*11+2)$ változatok egyaránt hibásak a sorban. Miért? Az első esetben nyilván az a hiba, hogy csak páros számok fognak előfordulni, például 7 soha nem jön ki. A másik változat hibája kevésbé szembetűnő, itt némi elméleti megfontolás segít. Könnyen látható, hogy például a kettes és hetes értékek nyilván nem egyformán valószínűek, hiszen előbbi csak 1+1 összegként adódhat, míg a másik $1+6, 2+5, 3+4, 4+3, 5+2, 6+1$ változatban is kijöhet. Pontosan az a helyzet, hogy egy nem egyenletes eloszlást kellene szimulálnunk, ez történik a 2. listában. A dolog „nem-egyenletes” voltát talán jól mutatja az alábbi táblázat:

	1	2	3	4	5	6
1	2	3	4	5	6	7
2	3	4	5	6	7	8
3	4	5	6	7	8	9
4	5	6	7	8	9	10
5	6	7	8	9	10	11
6	7	8	9	10	11	12

3. feladat: (De Méré lovag, XVIII. század). Próbáljuk meg eldönteni, mi a valószínűbb: egy kockával dobva négyszer egymás után, legalább az egyik hatos lesz; illetve, két kockával 24-szer dobva legalább egyszer 12 lesz az eredmény? Eddigi tudásunk alapján főleg a két kockával való dobás fogja megoldgatni a gépet: 100 kísérlet is unalmas

san sok időt igényel. A két valószínűség között viszont elég csekély különbség sejlik, ezért megint az elméletet hívjuk segítségül. Mindkét eseménynek az ellentettjét (ellenkezőjét) vizsgáljuk.

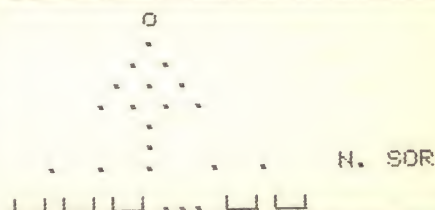
– Mennyi az esélye, hogy egyik dobás sem lesz hatos? Ez könnyen láthatóan $(5/6)^4$, amit kényelmi okokból géppel fogunk kiszámítani.

– Ugyanígy a 24 darab nem 12-es dobáshoz $(11/12)^{24}$ érték rendelkez, ami már kétségtelenül gépes feladat.

```
PRINT 1-(5/6)^4, 1-(11/12)^24
.517746913 .491403869
```

4. feladatként ide illeszthető a „születésnap-paradoxon” – ld BIT-LET

5. feladat: Szimuláljunk egy N-soros Galton-deszkát! Ez egy olyan falap, amibe az alábbiak szerint vertek szöveget:



Minden szögénél 1/2 eséllyel pattan a golyó jobbra vagy balra. Kérdés, hogyan töltődnek fel az utolsó sor alatti tálkák, ha K golyót engedünk útjára?

```
50 INPUT N,K
100 FOR I=1 TO K
120 A=0:FOR J=0 TO N
130 IF RND(1)>.5 THEN A=A+.5:GO TO A=A+.5
140 NEXT J
150 A=A+(N+1)/2:TR(A)=TR(A)+1
160 NEXT I
200 FOR J=0 TO N+1:PRINT TR(J);:NEXT J
```

Sajnos $N=10$ mellett a $K=1000$ esetet már alig győzzük kivárni. Próbáljuk gyorsítani a programot: ne minden szögénél egyenként döntsünk, hanem rögtön a tábla esés esélyeit vizsgáljuk. Segítenek a binomiális együtthatók!

Török Turul



K Ö N Y V M O L Y

Dr. Kovács Magda: Angol-magyar mikroszámítógép-mikroelektronikai szótár – LSI ATSZ, 253 o., 197 Ft

(A háromrészesre tervezett szótár-sorozat első kötete közel 5000 címszó magyar megfelelőjét adja meg, a szakemberek szövegmegértési és fordítói munkáját segítve.)

Sinclair Spectrum játék és program III. – LSI ATSZ, 175 o., 120 Ft
(A sorozat új kötete a játék-POKE-k, a mikrolexikon és néhány játék leírása után a teljes LASER programcsalád használatát mutatja be.)

Dr. Pajor Gábor: Az IBM PC-ről kezdő felhasználóknak I. A hardver – LSI ATSZ, 630., 66 Ft

(A munka az IBM gépekkel most ismerkedőknek nyújt segítséget: megismerteti az olvasót a számítógép fogalmával, és eligazítást nyújt a háttértárolók, a monitorok, a billentyűzetek és a nyomtatók között.)

IBM PC XT/AT felhasználói és technikai információs kártya – LSI ATSZ, 96 o., 126 Ft

(A kiadó információs kártya-sorozatához csatlakozó kötet a legelterjedtebb IBM gépek DOS-parancsai, BASIC-utasításai, valamint a gyakran használt gyári programok kezelésével ismerteti meg.)

Dr. Futó István: CPC BASIC három szinten – Műszaki Könyvkiadó, 383 o., 198 Ft

(A kötet a mind jobban terjedő Amstrad gépek kezelésével foglalkozik. Az első szint az általános tudnivalókat ismerteti, a második a gépek BASIC nyelvét, a harmadik pedig gépi kódú programozásukat – általános áttekintést nyújtva így a CPC gépek használatáról.)

Heift: CAD – DATA BECKER-Novotrade, 165 o., 290 Ft

(A könyv a Computer-Aided-Design, vagyis a számítógéppel segített műszaki tervezés alapjaiba vezeti be az olvasót, C64-re készített példaprogramok segítségével.)

Csikós Zsolt: C64/128 zenekedvelőknek – LSI-ATSZ, 144. o., 125 Ft

Az alábbiak olvashatók Csikós Zsolt kötetének előszavában: „Arra töre-

kedtem, hogy a könyv végére érve minden olvasó megbizonyosodjék: igenis lehet BASIC-ben is jó zene-programokat írni, ha az ember elég ügyes.”

A könyv áttanulmányozása alapján meggyőződhattunk róla, hogy ez valóban így van. A kötet tehát eléri célját, ami ekkora feladat kitűzése mellett nem kis érdem.

Hogy kinek szól a kötet? Az előszóban a szerző öt pontban sorolja fel a várható olvasói rétegeket, – és ebből az derül ki, hogy tulajdonképpen mindenkinek. Hiszen melyik Commodore-tulajdonos ne szeretne volna még kihasználni gépének állítólag csodálatos, ám BASIC-ből nehezen elérhető zenei lehetőségeit?

vassák el. Az első és a második fejezet a zeneprogramozás elméleti megalapozásul szolgál: az 1. a hangrendszerekkel és a kottaolvasással ismert meg (illetve a gyakorlottabbak itt átismételhetik a tudnivalókat), míg a 2. a C64 hanggenerálására szolgáló SID 6581 chip regisztereinek kezelését mutatja be.

A harmadik fejezetben kezdődik a dallamok programozása: itt a rövid, egyszólamú dallamok gépre vitelétől eljuthatunk a három hanggenerátort használó, többszólamú művekig. A fejezet nagy erőnye, hogy a szerző azt is leírja, hogyan is alakultak ki az egyes programok, illetve részletezi az egyszerűsítési lehetőségeket. Mint kuriózumot érdemes kiemelni a fejezet utolsó programját, ami egy ötszólamú darab két számítógépre készült feldolgozása.

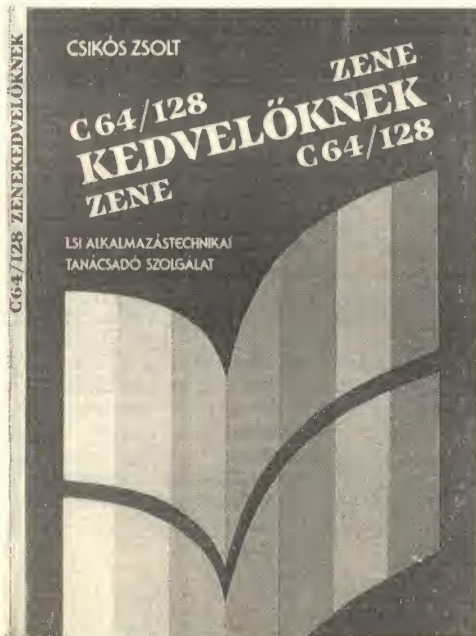
A negyedik rész egy jól rendszerezett szintetizátorprogramot mutat be, ez az eddigiek jó összefoglalásának tekinthető. Itt is nagyon jól követhető a program építése: ügyes trükkökkel fokozatosan növelhetjük a futás sebességét.

Az ötödik fejezet címe: Zene és grafika. Újra egyetlen hosszú programot készítettünk, amelyben az ötletes zenei megoldásokat SIMON'S BASIC-ben készült illusztrációk kísérik.

Az utolsó, hatodik fejezet egy elterjedt gyári program, a Music Shop lehetőségeit ismerteti. Persze, jóval egyszerűbb egy ilyen segédeszközzel zenélni, mint a hosszú DATA-sorokkal – ennek ellenére az előző fejezetekből megismertek sem haszontalan tudnivalók.

A Music Shop szolgáltatásainak ismertetésére azonban meglehetősen logikátlan sorrendben került sor, sőt néhány ki is maradt.

S itt kell rátérnünk a kötet fogyatékos-ságaira. A Music Shop bemutatása úgy szakad félbe, mintha elvágták volna. Itt véget is ér a kötet érdemi része – úgy tűnik, nem maradt elég idő a befejezésre. Ezt a félkész állapotot máshol is tapasztalhatjuk: így a szintetizátor előre programozható kilenc dallamából csak négyet találunk meg – mondván, hogy a többi nem



Jó felépítésű, logikus szerkezetű könyvecskét forgathatunk (egy-két hiányosságtól eltekintve – de ezekről később). Az előszót – szokatlan módon – a 0. fejezet követi, amely néhány lényeges tudnivalót tartalmaz a C64 csatlakoztatási lehetőségeiről és a programok beírásának lerövidítéséről. Ezek azért itt kaptak helyet, mert a szerző pesszimista meggyőződése szerint az előszót sokan úgysem ol-

PROGRAM CSERE-BERE

készült el a kötet nyomdába adásáig. Talán kevesebb sietséggel tökéletesebb könyvet sikerült volna kiadni. Még egy furcsaság: az utolsó fejezetet követő függelékek után az olvasó még négy értékes oldalra lapozhat rá: két dob-program rövid ismertetésére – de ezt a tartalomjegyzék sem jelzi, úgy tűnik, az utolsó pillanatban, „könyvzártakor” kerültek ide. Mindezek ellenére igen jól használható a kötet, és ezt segítik az egyes témakörökhöz kapcsolódó feladatok is – sőt a matematika iránt érdeklődő olvasó az egyes hullámformákhoz tartozó Fourier-sorok elemzésében is elmélyülhet a függelékben. Végül külön említést kell még tennünk a kötet kiviteléről: mind a szedést, mind az ábrákat, rajzokat Macintosh gépen készítették. A különböző betűtípusok jól tagolják a könyvet (bár meg kell jegyeznünk, hogy a fejezetcímek és alcímek zsinórírása kissé gyermeket), és elámulhatunk a gép csodálatos grafikai képességein.

Tallér József

POSTA

Alapproblémám az, hogy szeretnék egy számítógépet venni, de még nem sikerült eldöntennem milyen legyen. A Commodore 16 és a Commodore Plus/4 érdekel a legjobban. E két géppel kapcsolatban lenne három kérdésem:

1. A C16-ra írt gépi kódú programok futnak-e a Plus/4-en?
2. A Plus/4-ben van-e beépített MONITOR program?
3. A C16-os POKE utasításait megérti-e a Plus/4?

Slóvá Gábor, 2051 Biatorbágy, Kossuth u. 2.

Válaszaink: Igen-igen-igen! Háromszor is igen!

Tisztelt Szerkesztőség!

Nekem új C64-em van, de programoztam HT-n és ZX-en is, így nem esek gondba, ha a Commodore programok időszakosan háttérbe szorulnak, mert egy másfajta gépre írt program átírása C64-re jó szórakozást jelent. Ennek gyakorlása sokszor ütközik akadályba, mivel nincs birtokomban a gép részletes leírása, a Kernal-rutinok címei. Kérem önöket, hogy ha módjukban áll, küldjék el nekem, vagy a lapban közöljék kérésemet. Fáradozásukat előre is köszönöm.

Galambos Zsolt, 7200 Dombóvár, Ady E. út 1a. 1/3.

A levelében leírt táblázatokat elküldeni nem áll módunkban, időnként részleteket találhat lapunkban. Ennél sokkal részletesebb információhoz juthat a Data Becker-sorozat könyveiből (igaz, hogy az árak elég magasak). Memóriatérkép, ROM lista stb. pl. a „C64 belső felépítése” c. könyvben található. Ezek a könyvek szinte minden olyan könyvesboltban kaphatók, amelyek egyáltalán foglalkoznak műszaki vagy számítástechnikai könyvek árusításával.

Egy negatív tapasztalatomról szeretném tájékoztatni önöket röviden. A történet alapjául: hobbiszerűen foglalkozom a számítástechnikával, ill. programozással – az ÖTLET (és benne a BIT-LET), mivel Magyarországon nem dűskálhatunk az ilyen témájú sajtótermékekben, az egyik legfontosabb hírforrásom. Azt is itt olvastam, hogy másokkal egyetemben Lampért Gábor (Pápa, Beke J. u. 14.) programokat cserélne kazettán. Annak rendje-módja szerint küldtem egy programlistát, Gábor pedig az enyémmel együtt – amelyen már kijelölte a számára kívánatos programokat – elküldte az övét. Én a programokat tartalmazó kazettával együtt küldtem vissza Gábor lajstromát (kb. ugyanannyit jelöltem ki, mint ő). Telt, múlt az idő – kb. két hónap múlva írtam egy kissé sértődött hangú levelet a fiúnak, mire jött is vissza a kazetta a programok felével. Időhiányra hivatkozva elnézést kért, s a hiányzó dolgokat is kilátásba helyezte, de ehhez még egy kazettát kért.

Elküldtem...

Jelenleg is itt tart az ügy, pedig a második menet szeptemberben kezdődött, s levelet is írtam Gábornak. Ideje pedig bizonyosan van, hiszen újra hirdetett a januári BIT-LET-ben.

Nem tudom, egyébként milyen visszajelzéseket kapnak e fórum működéséről, ettől az egy alkalomtól eltekintve nekem is inkább öröm forrása, mint bosszúságé. Arra azonban figyelmeztetném az olvasótábor, hogy Lampért Gábor a kazettákat is gyűjti, nemcsak a programokat.

Üdvözlettel Bánhalmi Péter, Miskolc, Könyves K. u. 19. III/4.

Köszönjük, hogy megírta a történeteket. Mi csak egyet tehetünk: Lampért Gábor a Cserebere rovatból „diszkvalifikáljuk”.

C 64-er magnókazettás játékprogramokat cserélnék. Bíró István, 1025 Budapest, Felsőzöldmáli út 61–65.

Plusz 4 játék-, oktató- és felhasználói programokat cserélek, lemezen is. Kálmán Albert, 3300 Eger, Rákóczi út 31. III. 11. Telefonúzenet: 143-031, 330-345 (Bp.)

Elcserélném Primo 64-es gépet (ZX) Spectrum gépért vagy Commodore 16-os gépre. Schmidt Szilárd, 7900 Szigetvár, Széchenyi u. 43.

Sok programmal rendelkező, gépi kódban járatos **C 16 tulajdonost keresek**. Én is ilyen vagyok. Zalka Ernő, 9022 Győr, Árpád út 51/A

„Primo” játék- és egyéb programokat cserélek. Cím: Szarka Zoltán, 1185 Budapest, Duna u. 23.

Commodore VC 20-ra bővítő és alapgépes gépi kódú programokat cserélnék. Levélben vagy személyesen. Szabó Zoltán, 1211 Budapest, Táncsics M. út 65. I/4.

MSX rendszerű számítógépesek jelentkezését várom program- és tapasztalatcsere céljából. Tel.: (06-46)12-634 Jancsurák István, 3528 Miskolc, Dráva u. 7.

Commodore 16 programot cserélek! Bodorló Mihály, 5711 Gyula, Megyesi S. u. 271

Tudom, hogy mindenkinek szuverén joga az, hogy válaszol-e egy ismeretlen levelére vagy sem. Azt viszont már nem értem, hogy az olyan emberek miért adják közre a nevüket, címüket, akik a hozzájuk írt udvarias hangú levelet válasz nélkül hagyják. Ez történt velem a januári BIT-LET program cserebere kapcsán. Minden C16 címre írtam és ez ideig eredménytelenül. Szeretném egyszer kipróbálni, hogy vajon bírnám-e az iramot hasonló esetben, de azt megígérhetem, hogy minden levélre válaszolnék. Az se volna baj, ha mód lenne rá – címekeket kapnék és én írnék először. Fél éve, hogy van egy C16-os gépünk és gyerekeim (8 és 14 évesek) elég sok időt töltenek el mellette. Elsősorban nekik szeretnék kedvezni, hogy minél több programunk legyen, de a közösségben még van 9 gép és így nekik, ill. a használóiknak is tudnék segíteni.

Somogyi Gábor, 6527 Nagybaracska, Toldi u. 2.

NYERTESEK:

Megtartottuk a Quatroplus nyerő és a Harmadgépnyerő sorsolását.

A Quatroplust Csillag Péter budapesti, a Plus/4-est pedig Bíró Gábor kisvárdai pályázónk nyerte. Gratulálunk!

COMPUTER-S

ATARI NYERŐ

```

10 INPUT N
15 GOTO 10+(N>0)*(N=INT(N))*10
20 DIM T(N)
21 PRINT
25 I=0
30 I=I+1
35 INPUT T(I)
40 GOTO 30-(I=N)*15
45 I=0
46 PRINT
50 I=I+1
55 J=I
60 J=J+1
65 GOTO 85+(T(J)<T(I))*15
70 V=T(I)
75 T(I)=T(J)
80 T(J)=V
85 GOTO 60-(J=N)*30
90 PRINT T(I)
95 GOTO 50-(I=N-1)*50
100 PRINT T(N)
101 PRINT
105 Q$=INKEY$
110 GOTO 115+(Q$="")*10
115 GOTO 120-(Q$<>"K")*(Q$<>"I")
*(Q$<>"")*110
120 GOTO 130+(T(1)>0)*(T(1)<10)*
(T(1)=INT(T(1)))*5
125 GOTO 130+T(1)*10
130 PRINT "A LEGKISEBB SZÁM NEM
0 ES 10 KOZE ESO EGESZ!"
135 GOTO 235
140 PRINT "EGY", "ONE"
145 GOTO 235
150 PRINT "KETTO", "TWO"
155 GOTO 235
160 PRINT "HAROM", "THREE"
165 GOTO 235
170 PRINT "NEGY", "FOUR"
175 GOTO 235
180 PRINT "OT", "FIVE"
185 GOTO 235
190 PRINT "HAT", "SIX"
195 GOTO 235
200 PRINT "HET", "SEVEN"
205 GOTO 235
210 PRINT "NYOLC", "EIGHT"
215 GOTO 235
220 PRINT "KILENC", "NINE"
225 GOTO 235
230 PRINT "VISZONTLATASRA!"
235 END

```

Atari-nyerő, 1. feladatának megoldása:

Egy lehetséges program itt látható:

Megjegyzések: akit a számok betűvel való kiírásánál zavar az ékezet hiánya, az megcsinálhatja magyar helyett egy megfelelő más nyelven is a kiírást! A GOTO 235 sorok helyett END elfogadható, mi tartottuk magunkat a régebbi BASIC-ek azon kikötéséhez, hogy egy programban egy END utasítás lehet.

Atari-nyerő, 3. feladat

Mit csinál a lenti program?

Kérjük szöveggel leírni, hogy mire használható ez a program, pontosan mit és hogyan csinál és hogy vajon miért pont így? Aki nek van gyorsítási ötlete (hogyan lehetne viszonylag kis program-méret növeléssel gyorsabbá tenni a programot), azt is írja le! Megjegyzés: A program eléggé általános BASIC-ben íródott (mi C16-on futtattuk). Talán az egyetlen megjegyezni való, hogy az összehasonlítások értéke: -1, ha igaz, 0, ha nem. (90-es sor)

```

10 DIM T(2,20),E(41)
20 FOR I=1 TO 2
30 INPUT A$:L=LEN(A$):L(I)=(L+2)/3:IF L>60 THEN 30
EN 30
40 FOR J=1 TO L(I):T(I,J)=VAL(MID$(A$,L+3-3*J,3)):NEXT J
50 PRINT:PRINT:PRINT:NEXT I
60 FOR H=1 TO L(2):FOR J=1 TO L(1):E(J+H-1)=E(J+H-1)+T(1,J)*T(2,H):NEXT J,H
70 FOR J=1 TO 40:T=INT(E(J)/1000):Z=E(J)-1000
*T:E(J)=Z:E(J+1)=E(J+1)+T:NEXT J
80 K=-1:FOR J=41 TO 1 STEP -1:IFE(J)=0 AND K THEN EN
EXT J:PRINT:END
90 PRINT RIGHT$( " "+STR$(E(J)-(K=0)*1000),3
);:K=0:NEXT J:END

```

COMPUTER-S

Kérjük levelet és a levélre felragasztani! Beküldési határidő: 1987. június 30.